Flop Count for

Algorithm 7.1. Classical Gram-Schmidt (unstable)

for $j = 1$ **to** n		
$\quad v_j = a_j$		
$\quad$ **for** $i = 1$ **to** $j-1$	$(j-1)$ iterations	
$\qquad r_{ij} = q_i^H a_j$	$2m$ flops/inter. (approx.)	$4m(j-1)$ flops (approx.)
$\qquad v_j = v_j - r_{ij} q_i$	$2m$ flops/inter. (approx.)	
$\quad$ **end**		
$\quad r_{jj} = \| v_j \|_2$		$2m$ flops (approx.)
$\quad q_j = v_j / r_{jj}$		m flops
end		

Therefore, since each loop on j "costs" approximately

$$4m(j-1) + 3m$$

flops,the total flop count is approximately

$$\begin{aligned}
\sum_{j=1}^{n} (4m(j-1) + 3m) &= \sum_{j=1}^{n} 4m(j-1) + \sum_{j=1}^{n} 3m \\
&= 4m \sum_{j=1}^{n} (j-1) + 3m \sum_{j=1}^{n} 1 \\
&= 4m \sum_{j=1}^{n-1} j + 3m \sum_{j=1}^{n} 1 \\
&\doteq 4m \left(\frac{n^2}{2} \right) + 3m(n) \doteq 2mn^2
\end{aligned}$$

Flop Count for

Algorithm 7.2. Modified Gram-Schmidt

$$
\begin{array}{llll}
\textbf{for}\ \ i = 1\ \textbf{to}\ n & & & \\
\quad v_j \ = \ a_j & & & \\
\textbf{end} & & & \\
\textbf{for}\ \ i = 1\ \textbf{to}\ n & & & \\
\quad r_{ii} \ = \ \|\, v_i \,\|_2 & & & 2m\ \ \text{flops (approx.)} \\
\quad q_i \ = \ v_i / r_{ii} & & & m\ \ \text{flops} \\
\quad \textbf{for}\ \ j = (i+1)\ \textbf{to}\ n & (n-i)\ \ \text{iterations} & & \\
\qquad r_{ij} \ = \ q_i^H a_j & 2m\ \ \text{flops/inter. (approx.)} & \rbrace & 4m(n-i)\ \ \text{flops (approx.)} \\
\qquad v_j \ = \ v_j \ - \ r_{ij} q_i & 2m\ \ \text{flops/inter. (approx.)} & & \\
\quad \textbf{end} & & & \\
\textbf{end} & & &
\end{array}
$$

Therefore, since each loop on i "costs" approximately

$$4m(n-i) + 3m$$

flops, the total flop count is approximately

$$
\begin{aligned}
\sum_{i=1}^{n} (4m(n-i) + 3m) &= \sum_{i=1}^{n} 4m(n-i) + \sum_{i=1}^{n} 3m \\
&= \ 4m \sum_{i=1}^{n} (n-i) + 3m \sum_{i=1}^{n} 1 \\
&= \ 4m \sum_{i=1}^{n-1} i + 3m \sum_{i=1}^{n} 1 \\
&\doteq \ 4m \left(\frac{n^2}{2} \right) + 3m(n) \ \doteq \ 2mn^2
\end{aligned}
$$